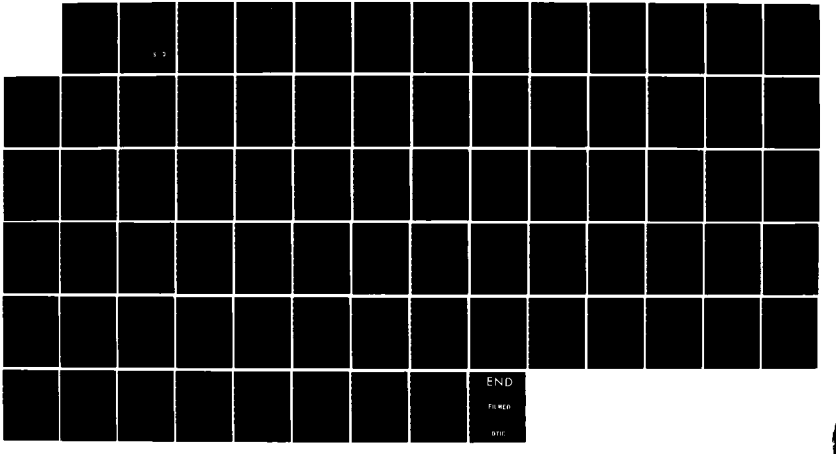
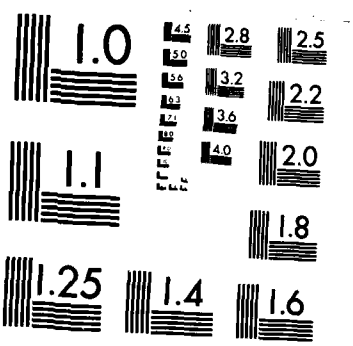
AD-A154 946 THE WEAPONS SUPPORT SYSTEM DATA SMOOTHING AND SPECTRUM ANALYSIS PROGRAM D.. (U) ANALYTIC SCIENCES CORP READING MA J V WHITE OCT 81 TASC-TR-1946-3 AFGL-TR-81-0306 1/1

UNCLASSIFIED F19628-80-C-0078 F/G 9/2 NL





AFGL-TR-81-0306

# THE WEAPONS SUPPORT SYSTEM DATA SMOOTHING AND SPECTRUM ANALYSIS PROGRAM DOCUMENTATION

James V. White

The Analytic Sciences Corporation
One Jacob Way
Reading, Massachusetts 01867

1 October 1981

Scientific Report No. 3



AIR FORCE GEOPHYSICS LABORATORY
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
HANSCOM AFB, MASSACHUSETTS 01731

## CONTRACTOR REPORTS

This technical report has been reviewed and is approved for publication.

THEODORE E. WIRTANEN
Contract Manager

THOMAS P. ROONEY
Chief, Geodesy & Gravity Branch

FOR THE COMMANDER

DONALD H. ECKHARDT
Director
Earth Sciences Division

This report has been reviewed by the ESD Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS).

Qualified requesters may obtain additional copies from the Defense Technical Information Center. All others should apply to the National Technical Information Service.

If your address has changed, or if you wish to be removed from the mailing list, or if the addressee is no longer employed by your organization, please notify AFGL/DAA, Hanscom AFB, MA 01731. This will assist us in maintaining a current mailing list.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|---|
| 1. REPORT NUMBER<br>AFGL-TR-81-0306 | 2. GOVT ACCESSION NO.<br>AD A154946 | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle)<br>THE WEAPONS SUPPORT SYSTEM DATA SMOOTHING AND SPECTRUM ANALYSIS PROGRAM DOCUMENTATION | | 5. TYPE OF REPORT & PERIOD COVERED<br>Scientific Report No. 3<br>6. PERFORMING ORG. REPORT NUMBER<br>TR-1946-3 |
| 7. AUTHOR(s)<br>James V. White | | 8. CONTRACT OR GRANT NUMBER(s)<br>F19628-80-C-0078 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>The Analytic Sciences Corporation<br>One Jacob Way<br>Reading, Massachusetts 01867 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS<br>320432AD |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>Air Force Geophysics Laboratory<br>Hanscom AFB, Massachusetts 01731<br>Monitor/Capt. Brian C. Mertz/LWG | | 12. REPORT DATE<br>October 1981<br>13. NUMBER OF PAGES<br>76 |
| 14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) | | 15. SECURITY CLASS. (of this report)<br>Unclassified<br>15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Gravity,
Gravity Data Evaluation,
Weapons Support System,
Data Editing,
Data Smoothing,
Spectrum Analysis,
Autoregressive Modeling.

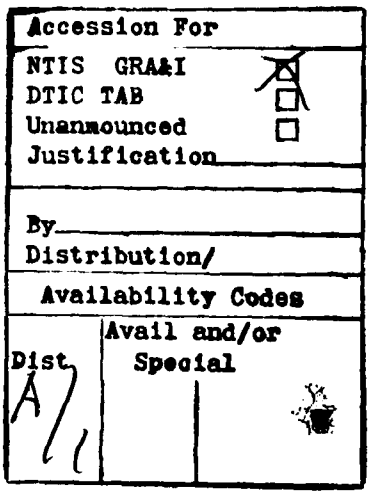
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report documents the four programs, GETBELL, AR, FFT, and PLOTFFT, which make up the data smoothing and spectrum analysis software supplied with the Weapons Support System. The programs are designed to run under the VAX/VMS operating system. Information regarding file structures and file-naming conventions is provided in the VAX/VMS System Reference Manuals.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)





## PREFACE

Documentation for the Weapons Support System (WSS), developed for the Defense Mapping Agency by the Analytic Sciences Corporation, consists of a User's Guide (TR-1946-1) that gives an overview of the WSS hardware and software, as well as detailed operating instructions for the major application areas listed below; and four volumes of detailed program documentation:

- *Gravity Data Evaluation Software* (TR-1946-2)
- Data Smoothing and Spectrum Analysis Programs (TR-1946-3)
- MULTISENSOR Simulation Software (TR-1946-4)
- GEOFAST Software Documentation (TR-1946-5)

This volume, Data Smoothing and Spectrum Analysis Program Documentation, documents the programs developed for these application areas, at a level suitable for use by programmers who will be adapting or modifying these programs. It is intended for use in conjunction with the Weapons Support System User's Guide and relevant operating system (VAX/VMS) documentation.

## TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

# 1. PROGRAM OVERVIEW

## 1.1 INTRODUCTION

This report documents the four programs, GETBELL, AR, FFT, and PLOTFFT, which make up the data smoothing and spectrum analysis software supplied with the Weapons Support System. The programs are designed to run under the VAX/VMS operating system. Information regarding file structures and file-naming conventions is provided in the VAX/VMS System Reference Manuals. The mathematical techniques referred to in this report are discussed in Ref. 1.

## 1.2 GENERAL SOFTWARE OVERVIEW

The programs GETDATA, AR, FFT, and PLOTFFT are intended for interactive analyses of two-channel time series data. The tasks of selecting data from master data files, smoothing and resampling the selected data, removing linear trends from the data, and plotting the data on the Lexidata graphics terminal or Printronix printer/plotter are all handled by program GETDATA. (The programs AR and PLOTFFT also have plotting capabilities.) Once selected data have been saved in data files by program GETDATA, they may be analyzed by running program AR, which does autoregressive modeling and power-spectrum analyses. Alternatively, the data may be analyzed by running program FFT, which smooths and resamples the master data file, divides it into separate batches and removes linear trends, windows each batch to reduce spectral leakage, and then computes power-spectrum estimates based on periodograms. The results of running program FFT are spectrum files that can be plotted by running program PLOTFFT.

In summary, GETDATA provides a graphical window through which master data sets may be viewed. GETDATA also creates new files of smoothed and resampled data that may be autoregressively analyzed by program AR. Programs FFT and PLOTFFT, in contrast, can be run as a complementary pair for operating directly on master data sets to compute and plot periodogram spectrum estimates.

## 1.3 DOCUMENT OVERVIEW

Section 2 of this report is a self-contained user's guide for the data smoothing and spectrum analysis programs. The user's guide explains how to run each program, how to interpret the prompting messages displayed on the VT100 monitor, and how to produce graphical plots on the Lexidata graphics terminal and hardcopy plots on the Printronix printer/plotter.

Section 3 contains detailed documentation for programmers who are maintaining the software. It describes the program organization, the data structures used, and the calling sequence of upper-level subroutines. It also lists the names and functions of all subroutines in the programs.

# 2. USER'S GUIDE TO THE SMOOTHING AND SPECTRUM ANALYSIS PROGRAMS

## 2.1 INTRODUCTION

There are four programs for the smoothing and spectrum analysis of time series data:

- Data smoothing and selection program GETDATA
- Autoregressive spectrum analysis and plotting program AR
- Periodogram spectrum analysis program FFT
- Periodogram spectrum plotting program PLOTFFT.

These programs are currently set up for analyzing two-channel time series (such as test data for gravity gradiometers that have two output channels). The program GETDATA is used to view selected portions of a master data set on the interactive graphics terminal. In addition, this program allows the user to prepare special data sets for later analysis via the program AR; in particular, the user may use GETADATA to smooth selected portions of a master data set, to resample the smoothed data at a selected rate, to view the results on the graphics terminal, and to save the smoothed and resampled data in a new file on disk.

After using the program GETDATA to plot data for analysis, the user has two choices for futher analysis of these data. The first choice is to create a file of selected data by using program GETDATA and then to do spectrum analysis of these data by running program AR. This choice uses both the

GETDATA and AR programs and yields autoregressive spectrum estimates. The second choice is to create a file of periodogram spectrum estimates by running program FFT and then to plot the spectrum estimates by running program PLOTFFT. This second choice uses the FFT and PLOTFFT programs and yields smoothed periodogram spectrum estimates.

The program AR computes an optimal autoregressive model for the process that generated the data and uses this model to plot the following quantities:

- Data being analyzed
- Sum-squared prediction errors of the AR model
- Autospectra for the two data channels
- Spectral coherence between the two data channels.

The program FFT smooths selected batches of data from a master data set, windows the batches to reduce spectral leakage, computes a periodogram spectrum estimate, and saves the results in a disk file for later use by the periodogram plotting program PLOTFFT.

The program PLOTFFT uses the data file created by FFT to plot the following quantities:

- Autospectra for the two data channels
- Spectral coherence between the two data channels.

## 2.2 DATA FORMATS

The programs are currently set up for two-channel data sets. This means that there should be two data values at each sample time in the master data set. With GETDATA, a maximum of 5000 two-channel samples can be selected for viewing from the master data set at one time.

The master data set should be a file with the following structure:

First Record = ASCII header describing the data set

(length equal to or less than 80 characters)

Example: BELL AF GGI S/N2 UMB ANGLE RUN 06/20/80

Other Records = pairs of data points in BINARY (2A4) format

2nd Record = 1st datum pair for channel 1 and channel 2

3rd Record = 2nd datum pair for channel 1 and channel 2

4th Record = 3rd datum pair for channel 1 and channel 2

5th Record = 4th datum pair for channel 1 and channel 2

*

*

*

The name of the master data set should preferably end in the suffix ".DAT"; e.g., "BELL1.DAT" is a valid data file name. (There are certain reserved system names that cannot be used as file names. You may wish to consult your systems manager for information about restrictions on data file names.)

## 2.3 DATA SELECTION (PROGRAM GETDATA)

To select data from a master data set, type RUN GETDATA. You will be asked for the name of the file containing the master data set. There are now three courses of action available to you:

1. Enter <RETURN> to choose the current default file
2. Type the name of a file using standard VAX format
3. Type QUIT to terminate the program. You may use this command whenever the program asks you a question.

You will now be asked for the number of smoothed data samples you wish to generate from the master data set. You may either enter <RETURN> to select the current default value displayed on the VT100 monitor, or you may type in a number of your choice followed by <RETURN>.

The next prompt asks for the location of the first datum you wish to select from the master data set. Typing 1 chooses the first datum in the master data set. You have the choice of either entering <RETURN> to select the current default value for the location, or typing a number of your choice.

The next prompt asks for the decimation factor. This number controls the degree to which the data are to be smoothed and resampled. For example, a decimation factor of 1 results in no smoothing and no resampling, while a value of 10 produces a selected set of data in which each sample is the average of 10 samples in the master data set. Therefore, in this case the sampling rate in the smoothed data set is one tenth the rate in the master data. You have the choice

The program continues by displaying a header that becomes the first part of the new data file. The first line of this header is a copy of the header in the master data set. The remaining lines list the decimation factor, the name of the file to be created, the number of data samples in the new file, the sampling rate of the smoothed data, the location of the first datum taken from the master data file in creating the current data set, and information about any linear trends (ramps) that might have been removed from the resampled data (the slopes of linear trends are expressed in data units per sample). You are now given the opportunity of typing a single line of additional information about the smoothed and resampled data. The maximum length of this line is 72 characters. Avoid using the dollar sign "$" because this symbol will cause all others following it to be ignored during certain printing operations. The valid characters are letters of the alphabet, integers, comma, period, and the following special symbols:

+ - * / ( ) =

You may use the <DELETE> key to correct typing errors before pressing <RETURN>. After <RETURN> is pressed, the program asks for a plot header. This is a short string of characters (length equal to or less than 22 characters) that will be printed automatically at the top of each plot. After you type the plot header, enter <RETURN>.

The program continues by creating the new data file on disk and then displaying

DO YOU WANT ANOTHER REGION OF DATA?

The program continues with the prompt

ENTER FRACTIONAL BANDWIDTH

The fractional bandwidth is a real number greater than or equal to zero that determines the frequency resolution of the spectrum estimates after the averaged periodogram is smoothed with a running mean. The width of the running mean is a fraction of its center frequency, and this fraction is the fractional bandwidth. This method of smoothing the periodogram yields constant-percent-bandwidth resolution, which is appropriate for the logarithmic frequency scales that are normally used for plotting spectra.

The program continues by processing the specified batches of data. During this processing, the means are subtracted from each windowed batch and then periodograms for each batch of data are computed with a mixed-radix Fast Fourier Transform (FFT) algorithm. These periodograms are averaged to produce a mean periodogram; the standard deviations of the periodograms are also computed if more than one batch of data is available.

The program continues by asking for the name of the file in which you wish to save the mean periodogram and standard deviations:

PLEASE ENTER THE NAME OF THE OUTPUT FILE

You may enter <RETURN> to choose the default file name, or type a file name of your choice. Examples of valid file names are BELL1 or BELL1.DAT. (The program will append the suffix ".DAT" automatically if you omit it.)

Entering <RETURN> selects the default option. Typing Y or N selects YES or NO options. Selecting option YES causes the program to subtract a zero-mean least-squares straight line from each channel of data. These lines are computed for each batch of data.

The program continues by displaying

ENTER KAISER-WINDOW ALPHA PARAMETER

You may enter <RETURN> to select the default value, or type a real number greater than or equal to zero. The Kaiser window is a bell-shaped weighting function. The program multiplies each batch of data by this weighting function to reduce spectral leakage in the periodograms. A value of zero for the alpha parameter yields a rectangular window (constant weighting). The larger the alpha parameter, the greater the attenuation of spectral leakage. However, larger values of alpha also yield lower frequency resolution in the periodogram. A value of 5.4414 has given satisfactory results in the analysis of gravity gradiometer self-noise data. The Kaiser window is defined in Ref. 1.

The program continues by asking for the number of seconds between adjacent samples in the master data set

ENTER SAMPLING INTERVAL

You may enter <RETURN>, which chooses the default sampling interval displayed on the VT100 monitor, or type a positive real number of your choice. If you do not know the correct sampling interval, you may type 1. The program will then continue to execute, but the spectral plots will be incorrectly labeled.

You may enter <RETURN> to choose the default number of batches, or type an integer of your choice. The program will attempt to segment the master data file into as many nonoverlapping adjacent batches as you specify. If there are insufficient data, the program forms as many batches as it can from the available data. These batches will contain the number of records you requested for each batch.

The program continues by displaying

ENTER STARTING RECORD

You may enter <RETURN> to select the default record, or type an integer. For example, if you type 1000, the program will skip the first 999 pairs of data in the master data file and start the first batch with record 1000. (Plots of the master data set can be generated interactively by running program GETDATA *before you run program* FFT.)

The program continues by displaying

ENTER DECIMATION FACTOR

You may respond by entering <RETURN> to choose the default decimation factor, or typing an integer. The value 1 results in no decimation (no smoothing and resampling). A value greater than unity causes the program to smooth the master data with a running mean and then to resample these smoothed data. The decimation factor is equal to the number of data points in the running mean, and it is also equal to the factor by which the sampling interval in the master data set is increased by the resampling process.

The program continues by displaying

DO YOU WANT A RAMP REMOVED FROM EACH BATCH?

Whenever the program asks you a question, you may terminate the program by typing

QUIT

The program starts by prompting for the name of the master data file that you wish to use:

ENTER INPUT FILE NAME

You may respond by entering <RETURN> to select the default file name displayed on the VT100 monitor, or you may type a file name of your choice.

The program responds by displaying the prompt

*ENTER NUMBER OF RECORDS IN A BATCH*

You may enter <RETURN>, which chooses the default value displayed on the VT100 monitor, or type a positive integer. This integer may be any integral power of 2 in the range from 2 to 4096, or it may be a highly composite number, such as 500, 1000, etc. If you choose a number that is <u>not</u> highly composite, such as 499, then the program will fail when the periodogram computation is attempted. (*This requirement for composite batch sizes is* inherent in the FFT algorithm.) The number of records in a batch is the number of consecutive data pairs from channels 1 and 2 of the master data file that are used to compute each periodogram. The larger this number, the greater the number of frequency bins in the periodogram.

The program continues by displaying

ENTER NUMBER OF BATCHES

These choices are self-explanatory except for choice F, which allows you to recompute the power spectra and coherence with a bandwidth and frequency resolution of your choice. The F option allows you to specify a minimum frequency limit, a maximum frequency limit, and the number of different frequencies at which the spectra and coherence are to be evaluated between these limits. The P, C, and F options cause the program to ask for minimum and maximum frequency limits of your choice. You may invoke autoscaling for these limits as follows: type 1 for both minimum and maximum frequency limits -- the program will respond by autoscaling the frequency scale. This technique of typing 1 for both minimum and maximum limits also invokes autoscaling when the program asks for y-axis limits in the P and C options.

The original time series data are not available for replotting once the power spectra have been computed.

## 2.5 PERIODOGRAM SPECTRUM ANALYSIS AND PLOTTING (PROGRAMS FFT AND PLOTFFT)

Any master data file having the format described in section 2.2 can be analyzed by running program FFT. This program estimates the power spectrum of the data by computing an averaged matrix periodogram from selected batches of windowed data in the master data file. Program FFT automatically saves the periodogram and its standard deviations in a disk file. After program FFT has created the periodogram file, you may plot the auto spectra and squared coherence on the Lexidata graphics terminal by running program PLOTFFT.

To run program FFT, type

```
RUN FFT
```

It is usually advisable to choose Y; later in the program you have the choice of overriding the autoscaling mode. (For illustrative purposes it is assumed that autoscaling is chosen.)

The program continues by computing the complex spectral density matrix for the best AR model. This matrix is computed at 300 logarithmically-spaced frequencies spanning the range from 0.3 times the reciprocal of the data length to the folding frequency (half the sampling frequency of the data being analyzed).

The program then continues by plotting the autospectra for both data channels on the Lexidata graphics terminal. The program displays the usual three-choice menu to give you the opportunity to print copies of the plots on the Printronix printer/plotter.

Entering <RETURN> or typing CL clears the screen. The program then continues by plotting the spectral coherence between the two channels of data, and the usual menu is displayed on the VT100 monitor. You may then choose to print the coherence function on the Printronix printer/plotter.

You may now replot the autospectra or the coherence with linear or logarithmic scales of your choice. This is done by entering <RETURN> or typing CL, causing a new menu to appear on the VT100 monitor:

P - PLOT POWER SPECTRUM

C - PLOT COHERENCE

F - CHANGE THE FREQUENCY RESOLUTION

Q - QUIT (terminates the program )

You have the choice of entering <RETURN>, which chooses the default value for model order, or you may type a number in the range of 1 through 40. The larger the order selected, the longer the computations take for fitting the models to the data. A maximum model order of 20 is reasonable for many data sets.

Once the maximum AR model order is selected, the program subtracts the means from both channels of data and then fits a family of AR models to them. The Akaike Information Criterion (AIC) is computed for each model and displayed on the VT100 monitor. The model order for which the AIC attains its algebraically smallest value is selected by the program. This is the model order that is best supported by the data for the purposes of modeling the process that generated the data. The program continues by plotting the sum-squared one-step-ahead prediction errors of the selected model for each data channel. If these plots are nearly straight lines, then the data have uniform statistical properties. On the other hand, if there are sudden large deviations from a straight line in these plots, then the locations of these deviations show places in the data set where there is exceptional behavior that is atypical of the rest of the data. These plots are useful for identifying outliers in nonhomogeneous data sets.

The menu described previously is now displayed; you have the option of continuing the program, printing the color graphics plot on the Printronix printer/plotter, or ending the program. When you continue the program by entering <RETURN> or typing CL, the following prompt appears on the VT100 monitor:

DO YOU WANT AUTO SCALING OF PLOTS?

You may respond by typing either Y or N.

terminal. A caption is plotted at the bottom of the display. The first line of the caption is the header from the master data file that was used in creating the data set currently being analyzed. The second and third lines contain the following information

- Decimation Factor (used in smoothing and resampling of data)
- Data Set (name of data file currently being analyzed)
- Number of Records (number of data samples currently being analyzed)
- Ramps (tells whether or not ramps were removed from the data and lists their slopes in data units per sample).

The fourth line in the caption contains any information that you previously typed into the header when the current data set was created by program GETDATA. This four-line caption will appear at the bottom of each plot generated by program AR.

After the program is finished plotting the data, it displays a menu on the VT100 monitor. The three options in this menu have the following meanings

<cr>/CLear - clears the Lexidata screen and continues the program

Print - prints a copy of the Lexidata plot on the Printronix printer/plotter

Quit - terminates the program

After you enter <RETURN> or type CL, the program displays the prompt

MAXIMUM AUTOREGRESSIVE MODEL ORDER (<41)

GETDATA are valid. In typing the name of the desired file, the suffix ".DAT" may be omitted; e.g., BELL1 is a valid file name. (There are certain reserved system names that can not be used as file names. You may wish to consult your systems manager for information about restrictions on data file names. Examples of invalid file names for this program are QUIT and EXIT.) Entering <RETURN> selects the default filename.

After reading the data from the selected file, the program displays the prompt

DO YOU WANT RAMPS REMOVED?

(You always have the option of terminating the program when a yes-no type questions like this one is asked; just type QUIT <RETURN>.)

To continue the program, you may respond by entering <RETURN>, which selects the default choice displayed on the VT100 monitor, or you may type Y or N. If you respond with Y, then a straight line is fitted to each channel of data by least squares and subtracted; the straight line is constrained to have an average value of zero so that the arithmetic means of the data in each channel are unchanged. The slopes of the straight lines are listed on the VT100 monitor and also listed in the captions of all graphs. The units of the slopes are data units per sample.

The program next displays the prompt

DO YOU WANT A PLOT OF THE DATA?

You may respond by typing Y or N. If you respond with Y, then both channels of data will be plotted on the Lexidata graphics

of each plot. After you type the plot header, enter <RETURN>. The program then creates the new file of smoothed and resampled data.

The next prompt asks if you want to start all over again and select more data from a master data file. You may respond with Y or N (or YES, NO). Typing Y causes the program to start over again. Typing N causes the program to terminate.

## 2.4 AUTOREGRESSIVE SPECTRUM ANALYSIS (PROGRAM AR)

Any data file created by GETDATA may be analyzed by running program AR, which plots the data, subtracts the mean from each channel of data and fits a family of autoregressive (AR) models to the residuals, selects the best model via the Akaike information criterion, and uses this model to generate prediction error, power spectrum, and coherence plots. The program AR is interactive and prompts the user at each step. In the following discussion, it is assumed for illustrative purposes that a data file named BELL1.DAT has previously been created by program GETDATA.

It is recommended that you start by typing DIR *.DAT <RETURN>. This causes a list of the data files in the current library to be displayed on the VT100 monitor for easy reference. For the present example, BELL1.DAT would be among the file names listed on the monitor. (If you do not want a list of the data files displayed on the VT100 monitor, then this step should be omitted.)

To run program AR, type RUN AR. The program AR will then respond with a prompt asking for the name of the data file that is be to analyzed. Data files created by program

automatically by the program to save your resampled data set for later analysis by the program AR. If you do not plan to use AR, then there is no need to use this file-creating option. The rest of this paragraph describes how to proceed should you wish to save your resampled data set. Enter <RETURN>, to select the default file name displayed on the VT100 terminal, or type a file name of your own choice. Examples of valid file names are BELL1 or BELL1.DAT. (The program will append the suffix ".DAT" automatically if you omit it.) The program then displays a header that becomes the first part of the new data file. The first line of this header is a copy of the header in the master data set. The remaining lines list the decimation factor, the name of the file to be created, the number of data samples in the new file, the sampling rate of the smoothed data, the location of the first datum taken from the master data file in creating the current data set, and information about any linear trends (ramps) that might have been removed from the resampled data (the slopes of linear trends are expressed in data units per sample). You are now given the opportunity of typing a single line of additional information about the smoothed and resampled data. The maximum length of this line is 72 characters. Avoid using the dollar sign "$" because this symbol will cause all others following it to be ignored during certain printing operations. The valid characters are letters of the alphabet, integers, comma, period, and the following special symbols

+ - * / ( ) =

You may use the <DELETE> key to correct typing errors before pressing <RETURN>. After <RETURN> is pressed, the program asks for a plot header. This is a short string of characters (length $\leq$ 22 characters) that will be printed automatically at the top

The program next generates plots of the smoothed and resampled data on the Lexidata graphics terminal, provided that the Lexidata terminal is available. The plot is for channel 1.

A menu is now displayed on the VT100 monitor; you may select any of the three possibilities listed there by typing the command you select. As a shortcut, you may type only the capitalized portion of the command. In this menu, <cr> is an abbreviation for <RETURN>. The commands are used for the following purposes:

<cr>/CLear - to clear the Lexidata screen and continue to the plotting of the data in channel 2

Print - to print a copy of the graph displayed on the Lexidata screen

Quit - to terminate the program

The abbreviations CL, P, and Q are valid abbreviations for CLEAR, PRINT, and QUIT.

After the data for channel 2 have been plotted on the Lexidata terminal, the menu is again displayed. This time when <cr>/CLear is selected, the program responds by asking whether you wish to save the data in a disk file. You may respond by typing YES, NO, or QUIT. The abbreviations Y, N, and Q are valid.

Typing Q will terminate the program. Typing N will cause the program to ask if you wish to continue by selecting some more data from the master data set (typing NO or N in response to this question will terminate the program). Typing Y will cause the program to ask for the name you want to use for the output file. This is the file that will be created

of either entering <RETURN> to select the default value for the decimation factor, or you may type a number of your choice.

The next prompt asks you for the sampling interval. This is the number of seconds between adjacent samples in the master data set. If you do not know the correct value for this quantity, you may type 1; the program will then continue to run and you will be able to view the data, but the header information that is generated for the selected data may be incorrect. You have the choice of either entering <RETURN> to select the current default value of the sampling interval, or you may type a number of your own choice.

The next prompt asks for the name of the dimensional units for the data being analyzed. For example, the correct response to this prompt for gravity gradiometer data is EOTVOS. As usual, you may simply enter <RETURN> to select the current default value for data units displayed on the VT100 monitor.

The program now responds by reading the master data file, listing the ASCII header from the master file, and printing the number of smoothed and resampled data points that have been produced during the reading operation. The number of resampled data points may be smaller than the number you requested; this happens when there are not enough data in the master data file to satisfy your request.

The next prompt asks whether you want linear trends removed from each channel of the processed data set. Responding with Y (or YES) to this prompt causes a best-fit least-squares straight line of zero mean value to be subtracted from each channel of resampled data. You may respond by entering <RETURN> for the current default setting, or you may type either N (or NO) to defeat the straight-line subtraction.

This allows you either to terminate the program by typing N, or to start the program over again by typing Y. You may alternatively enter <RETURN> to select the default option displayed on the VT100 monitor.

To plot the periodogram spectra on the Lexidata graphics terminal, you should run program PLOTFFT by typing

RUN PLOTFFT

Whenever the program asks you a question, you may terminate the program by typing

QUIT

Program PLOTFFT starts by displaying

ENTER INPUT FILE NAME

Any file created by program FFT is valid. You may either enter <RETURN> to select the default file name, or you may type a file name of your choice.

The program continues by displaying a four or five line header that describes the data set selected for analysis. The first line is the header from the master data set from which the current data set was derived. The second line gives the decimation factor and the name of the current data set. The third line lists the number of batches, number of samples per batch, the sampling interval, and the location (NSTART) of the first datum taken from the master data file in creating the current data set. The fourth line lists the Kaiser-window alpha parameter and the fractional bandwidth. The last line may be blank or may contain a descriptive message about the data.

The program continues by plotting the auto spectra on the Lexidata graphics terminal. (The plotting scales are auto-scaled. The solid lines are the estimated spectra; the dotted lines are standard errors of these estimates.) A menu is displayed on the VT100 terminal, which gives you the option of typing

| | |
|---|---|
| <return> | to clear the Lexidata screen and continue |
| P<return> | to produce a copy of the graphics on the printer/plotter |
| Q<return> | to terminate the the program |

The program continues by plotting the estimated spectral coherence between the two data channels and the standard errors of the estimate (dotted lines). The menu is displayed to give you the opportunity of selecting a hard copy of the graphics.

The program continues by displaying a new menu that allows you to change the plotting parameters

P - PLOT POWER SPECTRUM

C - PLOT COHERENCE

F - CHANGE THE FREQUENCY RESOLUTION

Q - QUIT (terminates the program )

These choices are self-explanatory except for choice F, which allows you to replot the power spectra and coherence with a different amount of smoothing applied to the periodogram. Typing F results in the prompt

ENTER FRACTIONAL BANDWIDTH

You may enter <RETURN> to select the default bandwidth, or type a positive fraction of your choice. The smaller the fractional bandwidth, the greater the frequency resolution of estimated spectra. However, smaller bandwidths also increase the standard errors of the estimated spectra. Therefore, some experimentation is needed to choose a fractional bandwidth that yields the best tradeoff between resolution and stability of the estimated spectra. Typical values of the fractional bandwidth lie in the range of 0.1 to 0.3.

If you type either P to replot the power spectrum, or C to replot the coherence, then the program asks for the type of scales for the X (horizontal) and Y (vertical) axes of the plots. The choices include all possible combinations of linear and logarithmic scales.

The program continues by asking you for the minimum and maximum frequency limits. You may enter <RETURN> to select the default value, or type a number of your choice. To invoke autoscaling, type 1 for both the minimum and maximum frequency limits. Autoscaling for the y-axis scales may also be invoked by typing 1 for both minimum and maximum limits.

The program continues by plotting the auto spectra or coherence on the Lexidata graphics terminal. After the plotting is finished, a menu appears on the VT100 monitor to give you the choice of printing a copy of the graphics on the plotter, continuing, or quitting. Typing Q terminates the program. Typing P causes the graphics to be printed on the plotter. Entering <RETURN> or typing CL clears the screen and continues the program by returning to the menu of plotting options.

## 2.6 ERROR AND DIAGNOSTIC MESSAGES

The smoothing and spectrum analysis programs have many built-in error traps to flag erroneous data supplied by the user. When you are promped to select an option from a menu, the program checks the validity of any selected option and reprompts you if an invalid option is requested. The program also checks for consistency in the plotting limits and reprompts if some of the values are invalid, such as a zero value to be plotted on a logarithmic scale.

The following message will appear on the screen if the program cannot open and read a data file you specified:

ERROR IN OPENING DATA FILE

The two most likely causes of this error are that you misspelled the file name, or that the file you specified is not compatible with the program you are running. For example, programs GETDATA and FFT can read master data files, while program AR is designed to read data files created by GETDATA, and program PLOTFFT is designed to read data files created by program FFT.

Program AR is intended for analyzing bivariate (two-channel) time series. The program will fail if the time series is degenerate (i.e., one of the channels is identically zero or if both channels contain identical data).

# 3. DETAILED PROGRAM DESCRIPTIONS

## 3.1 GETDATA PROGRAM

GETDATA performs the data selection phase for the autoregressive analysis program (AR). GETDATA may also be used to view and plot data from a master data file, (before processing the data, for example, with program FFT). A block flow chart of the subroutine structure is given in Fig. 3.1-1. Some of the subroutines are common to both phases (GETDATA and AR). Section 3.7 gives a brief programmer-oriented description of each subroutine, including function and input/output variables.

The GETDATA program consists of a main program and a library containing the subroutines, named ARLIB.OLB. The main program is a driver routine that reads default file names and parameter input from a datastream file, and sets default parameters for the default data set. The name of the datastream file is GETDATA.DEF and can be modified to fit the new data sets. The following is an example of the default values for a BELL gradiometer:

```
NRECS   = 1000           !number of records in output
                          data set
NDEC    = 1              !decimation factor
NSTART  = 1              !starting record number
SINT    = 2.0            !sampling interval in seconds
UNITS   = 'EOTVOS'       !data units
CSLOPE  = 'NO'           !slope removal flag
NMAX    = 5000           !maximum number of output
                          records
FNAME   = 'BELLUMB.DAT'  !input master data file name
ONAME   = 'TEMPAR'       !output selected data set name
;                        !end of input stream
```

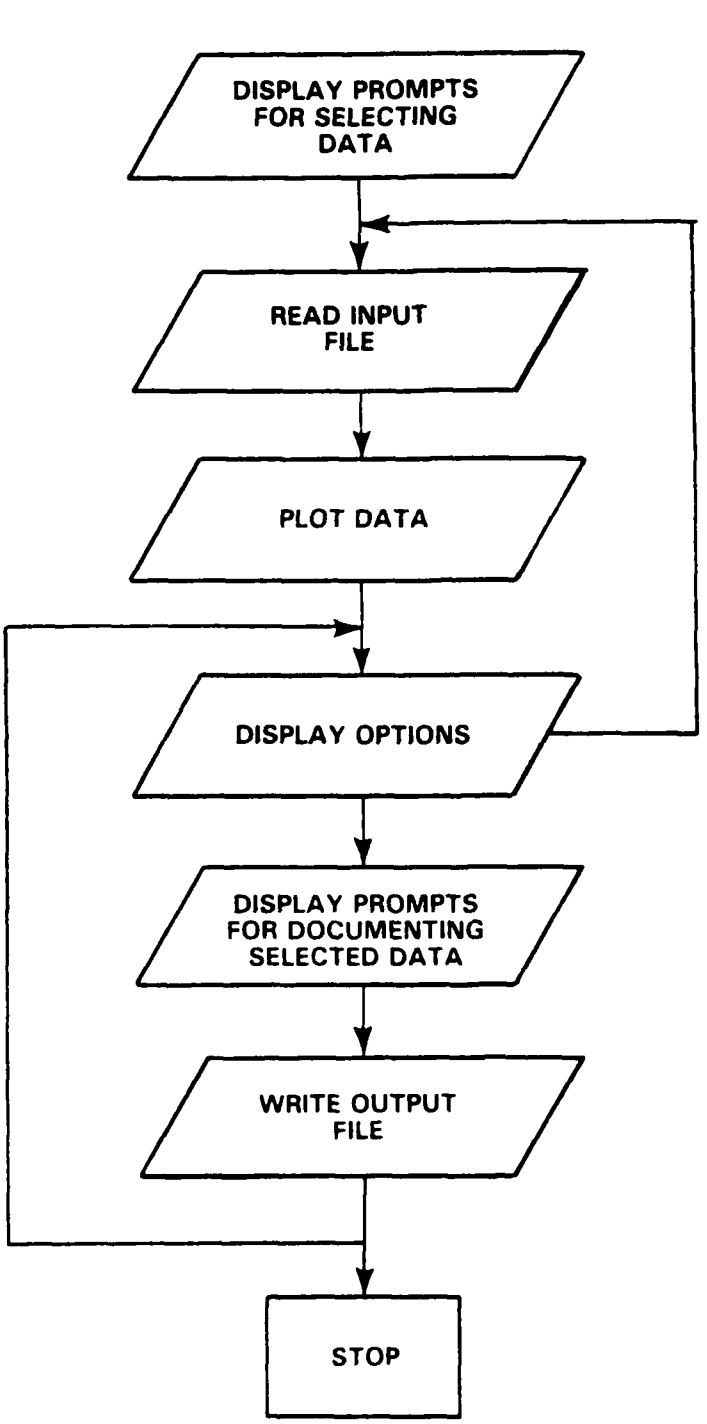



Figure 3.1-1 Block Flow Chart of GETDATA

The maximum size of an output data set is fixed at 5000 points. To increase this size it is necessary to increase the array dimensions to the desired size in the main program and then recompile and link the program. To recompile and link GETDATA the user enters the following commands:

```
FORTRAN GETDATA      !compiles main program
@LG                  !links program and creates new
                     !load module
```

If changes in any of the subroutines are necessary then the following command sequence can be used to create a new version of GETDATA from the updated source:

```
@CL 'program name'   !compiles and loads into ARLIB
@LG                  !links program and creates new
                     !load module
```

Basic program information in a standard format is given below.

Name: GETDATA

Function: This main program creates and plots data files from master data files

Common Blocks: *NONE*

Subprograms called: NMREAD,INPCHR,INPIN4,INPRL4,PRTINF,JREMOV, ANOTAT,EZXY,DSCLS

## 3.2 AR PROGRAM

AR performs autoregressive modeling and plotting functions for autoregressive spectrum estimation. A block flow chart of the subroutine structure is given in Fig. 3.2-1. Some

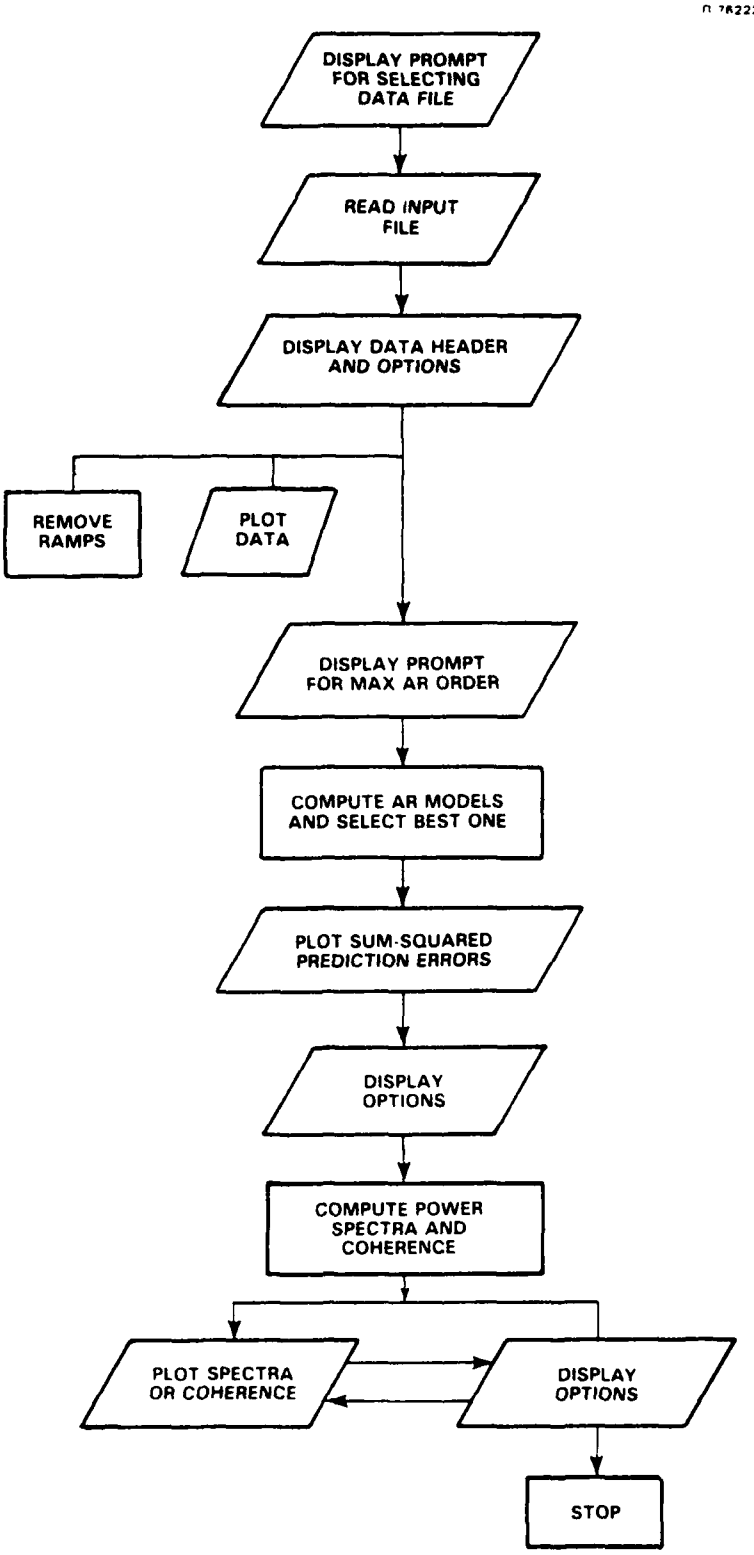



Figure 3.2-1 Block Flow Chart of AR

of the subroutines are common to both GETDATA and AR. The mathematical algorithms involved are discussed in Ref. 1. Section 3.7 gives a brief programmer-oriented description of each subroutine, including function and input/output variables.

The AR program consists of a main program and a library containing the subroutines, named ARLIB.OLB. The main program is a driver routine that reads default file names and parameter input from a datastream file, and sets default parameters for the default data set. The name of the datastream file is AR.DEF and can be modified to fit the current data set. The following is an example of the default values for a BELL gradiometer:

```
NBINS      = 300          !number of frequency bins
MAXORDER   = 20           !maximum autoregressive order
                           used
NDIM       = 2            !number of data channels
IPLT       = 'YES'        !raw data plot flag
AUTOSCAL   = 'YES'        !automatic plot scaling flag
RAMP       = 'NO'         !ramp removal flag
;                         !end of input stream
```

The maximum size of the input data set is fixed at 5000 points. To increase this size, it is necessary to increase the array dimensions to the desired amount in both the main program and JPLOT; then recompile and link the programs. To recompile and link the programs, the user enters the following commands:

```
FORTRAN AR              !compiles main program
@CL JPLOT               !compiles and loads JPLOT into
                         ARLIB
@LG                     !links program and creates new
                         load module
```

If changes in any additional subroutines are found to be necessary then the following command sequence can be used to create a new version of AR using the updated source:

```
@CL 'program name'    !compiles and loads into ARLIB
@LG                   !links program and creates new
                       load module
```

Basic program information in a standard format is given below.

Name: AR

Function: Autoregressive modeling of time series and spectrum estimation

Common Blocks: WKSPAC

Subprograms called: NMREAD,INPCHR,PRTINF,JREAD,AGGETP,J8REM, JPLOT1,INPIN4,JEQUAT,MMAXEN,JSUMRD,CPOWER, INPRL4,DSCLS

## 3.3 FFT PROGRAM

FFT performs the data selection and periodogram computations for use by the periodogram spectrum plotting program (PLOTFFT). A block flow chart of the subroutine structure is given in Fig. 3.3-1. Some of the subroutines are common to both FFT and PLOTFFT. A discussion of the mathematical algorithms involved is presented in Ref. 1. Section 3.8 gives a brief programmer-oriented description of each subroutine, including function and input/output variables.

The FFT program consists of a main program and a library containing the subroutines named PERIODLIB.OLB. The main program is a driver routine that reads default file names and parameter input from a datastream file, and sets default parameters for the default data set. The name of the datastream file is FFT.DEF and can be modified to fit the current data set. The following is an example of the default values for a BELL gradiometer:

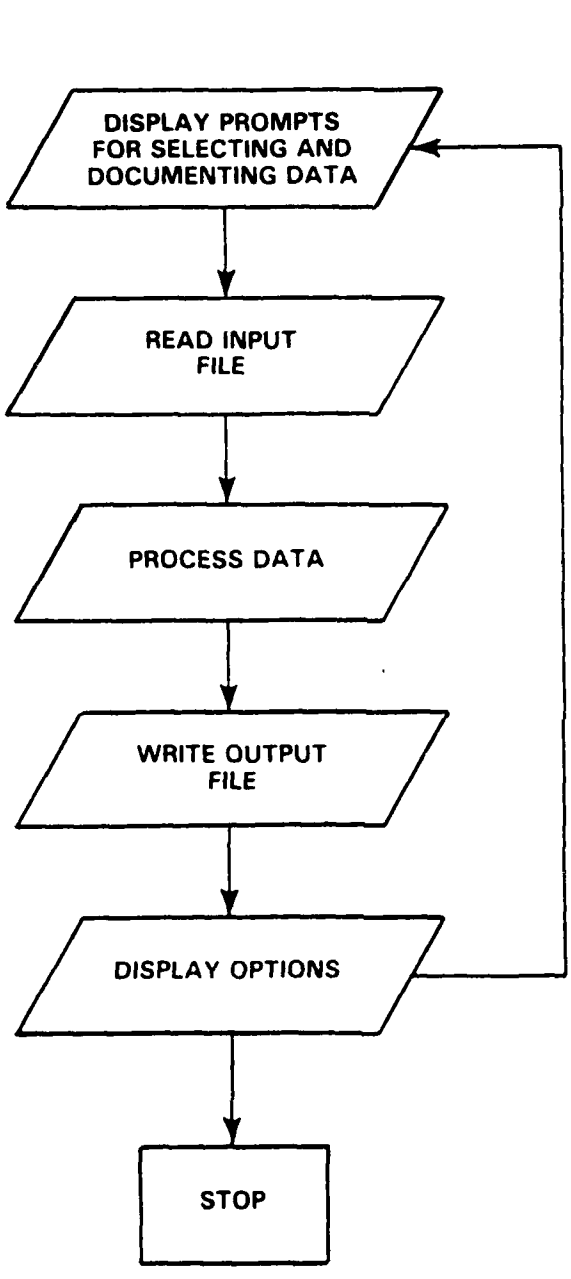



Figure 3.3-1 Block Flow Chart of FFT

```
CSLOPE = NO              !slope removal flag
NSIZE  = 1024            !size of individual batches
NBATCH = 20              !number of batches
NSTART = 1               !starting record number in master data file
FRACB  = 0.1             !fractional bandwidth
ALPHA  = 5.4414          !Kaiser data-window parameter
SINT   = 2.E0            !sampling interval in seconds
NDEC   = 1               !decimation factor
IN FILE = 'BELLUMB.DAT'!input file name
OUT FILE = 'TEMPFILE'    !output file name
;                        !end of input stream
```

The maximum size of a batch is fixed at 4096 points. To increase this size it is necessary to increase the array dimensions to the desired size in the main program and then recompile and link the program. To recompile and link FFT, the user enters the following commands:

```
FORTRAN FFT              !compiles main program
@FFT                     !links program and creates new
                         !load module
```

If changes in any of the subroutines are necessary then the following command sequence may be used to create a new version of FFT from the updated source:

```
@CL 'program name'       !compliles and loads into arlib
@FFT                     !links program and creates new
                         !load module
```

Basic program information in a standard format is given below.

Name: FFT

Function: Estimates power spectra and coherence from two-channel time series via periodogram analysis

Common Blocks: TRIGS,NNTEST

Subprograms called: NMREAD,INPCHR,INPIN4,PRTINF,INPRL4,RDATA, SMEAN,RSLOPE,KWINDO,FFTKW,REALVEC,FMEAN, STDEV

## 3.4 PLOTFFT PROGRAM

PLOTFFT plots the power spectra and coherence data produced by the FFT phase of the power spectrum analysis. A block flow chart of the subroutine structure is given in Fig. 3.4-1. Some of the subroutines are common to both phases (FFT and PLOTFFT). A discussion of the mathematical algorithms involved is presented in Ref. 1. Section 3.8 gives a brief programmer-oriented description of each subroutine, including function and input/output variables.

The maximum size of the input data set is fixed at 4096 points. To increase this size it is necessary to increase the array dimensions to the desired size in the main program PLOTFFT and subroutines SMFFTS,SMPLOTS, and PPLOT; then recompile and link the programs. To recompile and link PLOTFFT one enters the following commands:

```
FORTRAN PLOTFFT      !compiles main program
@CL SMFFTS           !compiles and loads SMFFTS into
                      PERIODLIB
@CL SMPLOTS          !compiles and loads SMPLOTS into
                      PERIODLIB
@CL PPLOT            !compiles and loads PPLOT into
                      PERIODLIB
@PLOTFFT             !links with main program and
                      creates new load module
```

If changes in any additional subroutines are found to be necessary then the following command sequence can be used to create a new version of PLOTFFT from the updated source:

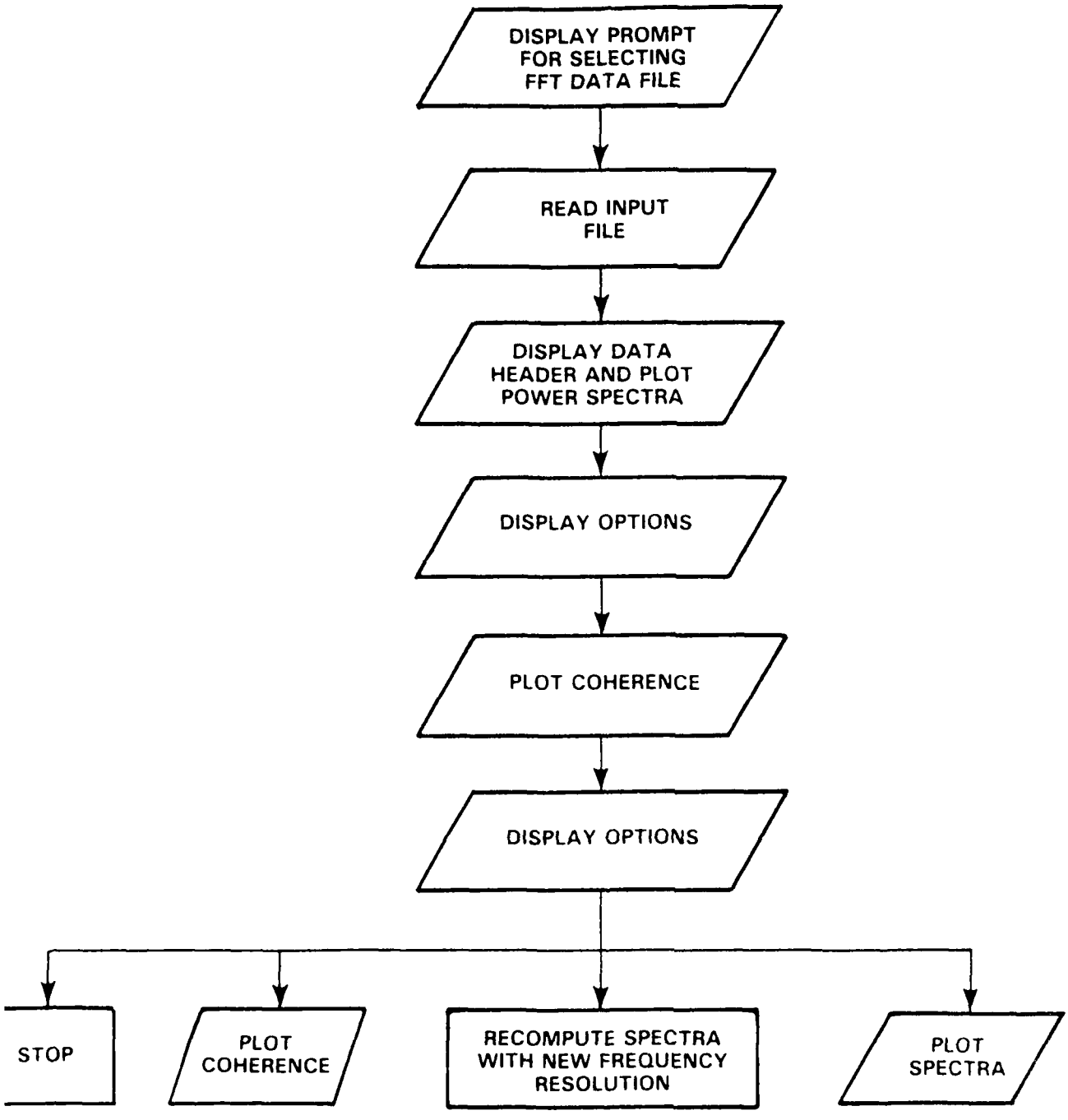



Figure 3.4-1 Block Flow Chart of PLOTFFT

### 3.7.16 Subroutine MMAXEN

CALLING SEQUENCE:

```
MMAXEN(N,ND,M,X,IA,V,A,B,PF,PB,Y,U,W,C,D,E,NE,AICVEC)
```

FUNCTION: Multivariate maximum entropy stepwise estimation of autoregression prediction matrices and one step prediction covariance matrix with order selection by Akaike's information criterion

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| N | I*4 | - | Input | Sample size |
| ND | I*4 | - | Input | Dimension of multivariate time series |
| M | I*4 | - | Input | Maximum order autoregession searched for best fit |
| X | R*8 | (*,*) | Input | Matrix of time series data |
| Y | R*8 | (*,*) | Input | Work matrix with same dimension as X used for residuals from backward predictions. Residuals from forward prediction are stored in X. Original data are destroyed. |
| W | R*8 | (*,*) | Input | Work matrix (ND,ND) |
| C | R*8 | (*,*) | Input | Work matrix (ND,ND) |
| D | R*8 | (*,*) | Input | Work matrix (ND,ND) |
| E | R*8 | (*,*) | Input | Work matrix (NE,NE) |
| NE | I*4 | - | Input | First dimension of matrix E in calling program (NE=ND*ND) |
| IA | I*4 | - | Output | Order selected by Akaike's information criterion |
| V | R*8 | (*,*) | Output | Estimated one step forward prediction covariance matrix |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.7.15 Subroutine JSUMRD

CALLING SEQUENCE:

JSUMRD(YS,XS,IA,X,A,N,HEADER,IHD,HDINFO,RDLAB)

FUNCTION: This routine calculates the sum-squared residuals

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| YS | R*4 | (0:*,2) | Input | Plot work array |
| XS | R*4 | (5000) | Input | Plot work array |
| IA | I*4 | - | Input | Order selected by Akaike's information criterion |
| X | R*8 | (*,2) | Input | Original data arrays |
| A | R*8 | (2,2,40) | Input | Estimated partial forward auto-regression matrices |
| N | I*4 | - | Input | Number of samples |
| HEADER | CHAR | - | Input | Plot header line |
| IHD | I*4 | - | Input | Number of plot identification lines |
| HDINFO | CHAR | (5) | Input | Plot identification lines |
| RDLAB | CHAR | - | Input | Ordinate label for plots |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: DISPLA,AGSETP,ANOTAT,EZXY,PWRY,FRAME

COMMON BLOCKS:

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| X | R*8 | (*,2) | Output | Contains data which were read in |
| N | I*4 | - | Output | Number of data points in X |
| IFLAG | I*4 | - | Output | = 0 - Successful read<br>= 2 - Data set did not contain enough points |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.7.14 Subroutine JREMOV

CALLING SEQUENCE:

JREMOV(X,NSIZE,HDINFO,SLOPE)

FUNCTION: Removes the slope from both channels found in X

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| X | R*4 | (5000,2) | Input | Raw data to have slopes removed |
| NSIZE | I*4 | - | Input | Number of points in raw data |
| HDINFO | CHAR | (5) | Output | Slope information will be added to HDINFO |
| SLOPE | R*4 | (2) | Output | Slope values returned to main program |
| X | R*4 | (5000,2) | Output | Return data with slopes removed |

### 3.7.12 Subroutine JPLOT1

CALLING SEQUENCE:

JPLOT1(X,Y,N,HEADER,HDINFO,IHD,RDLAB)

FUNCTION: Generates plots of the selected data

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| X | R*8 | (*,2) | Input | Input double precision data arrays (N,2) |
| Y | R*4 | (*,2) | Input | Input temporary single precision data arrays (N,2) |
| N | I*4 | - | Input | Size of data arrays to be plotted |
| HEADER | CHAR | - | Input | Header associated with input data |
| HDINFO | CHAR | (5) | Input | Additional information carried along with the input data |
| IHD | I*4 | - | Input | Number of HDINFO lines |
| RDLAB | CHAR | - | Input | Label along the ordinate axis |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: DISPLA,ANOTAT,AGSETP,EZY,PWRY,FRAME

### 3.7.13 Subroutine JREAD

CALLING SEQUENCE:

JREAD(X,N,IFLAG)

FUNCTION: Routine to read input data to the AR modeling program

### 3.7.11 Subroutine JPLOT

CALLING SEQUENCE:

JPLOT(S,NIN,F,IA,HEADER,IHD,HDINFO,ENDFLG,PSLABEL,AUTOFG)

FUNCTION: This routine plots the power spectrum for each channel of data and the coherence between channels

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| S | C*16 | (2,2,*) | Input | Power spectral density matrix for each frequency bin |
| NIN | I*4 | - | Input | Number of frequency bins |
| F | R*8 | (*) | Input | Array of frequencies at which power spectral densities are calculated |
| IA | I*4 | - | Input | Order selected by Akaike's information criterion |
| HEADER | CHAR | - | Input | Plot frame header |
| IHD | I*4 | - | Input | Number of plot description lines |
| HDINFO | CHAR | (5) | Input | Plot description lines |
| PSLABEL | CHAR | - | Input | Ordinate axis label for power spectral density plot |
| AUTOFG | I*4 | - | Input | = 1 - Use automatic scaling<br>= 0 - Prompt for axis limits |
| ENDFLG | I*4 | - | OUTPUT | Set by user to terminate the program |

COMMON BLOCKS: WKSPAC

SUBPROGRAMS CALLED: AGSETP,INPRL4,DISPLA,ANOTAT,AGSETF,EZXY, PWRY,FRAME,INPIN4

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| X | R*8 | (*,2) | Input | Contains original input data |
| NSIZE | I*4 | - | Input | Number of points in input data set |
| X | R*8 | (*,2) | Output | Raw data with slopes removed |
| HDINFO | CHAR | (5) | Output | Header lines with slope information added to be carried along with the data file |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.7.10 Subroutine JEQUAT

CALLING SEQUENCE:

JEQUAT(X,XX,NSIZE)

FUNCTION: This routine equates matrix XX to matrix X

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| X | R*8 | (*,2) | Input | Input matrix(NSIZE,2) |
| NSIZE | I*4 | - | Input | Size of matrix X and matrix XX |
| XX | R*8 | (*,2) | Output | Equals matrix X upon return to calling program |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLEDS: *NONE*

### 3.7.8 Subroutine INVERT

CALLING SEQUENCE:

INVERT(B,N,ND)

FUNCTION: Inverts N by N matrix B in place; ND is first dimension of B in calling program

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| B | R*8 | (*,*) | Input | Matrix to be inverted (N,N) |
| N | I*4 | - | Input | Dimension of matrix B |
| ND | I*4 | - | Input | First dimension of B in calling program |
| B | R*8 | (*,*) | Output | Resultant inverted matrix |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.7.9 Subroutine J8REM

CALLING SEQUENCE:

J8REM(X,NSIZE,HDINFO)

FUNCTION: This routine removes slopes from both channels of input data X

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| IA | I*4 | - | Input | Order selected by Akaike's information criterion |
| A | R*8 | (2,2,40) | Input | Estimated partial forward autoregression matrices |
| SAMPT | R*8 | - | Input | Sampling interval |
| AA | C*16 | (2,2) | Input | Work array |
| V | R*8 | (2,2) | Input | Estimated one step forward prediction covariance matrix |
| F | R*8 | (*) | Input | Array of frequencies at which power spectral densities are to be calculated |
| NIN | I*4 | - | Input | Number of frequency bins |
| S | C*16 | (2,2,*) | Output | Power spectral density matrix for each frequency bin |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: CMZERO, CMI2X2,CMMULT,CMCTR,CMSCAL

### 3.7.7 Subroutine DET

CALLING SEQUENCE:

DET(B,A,N,D)

FUNCTION: Evaluates determinant of an N by N matrix B

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| B | R*8 | (*,*) | Input | N by N input matrix |
| A | R*8 | (*,*) | Input | N by N work matrix |
| N | I*4 | - | Input | Dimension of B and A |
| D | R*8 | - | Output | Determinant of matrix B |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.7.5 Subroutine CMZERO

CALLIN SEQUENCE:

```
CMZERO (A,N1,N2)
```

FUNCTION: This routine zeroes out a complex matrix

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| A | C*16 | (*,*) | Input | Complex matrix to be zeroed out (N1,N2) |
| N1 | I*4 | - | Input | First dimension of A |
| N2 | I*4 | - | Input | Second dimension of A |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.7.6 Subroutine CPOWER

CALLING SEQUENCE:

```
CPOWER(IA,A,SAMPT,AA,S,V,F,NIN)
```

FUNCTION: This routine calculates the power spectrum density matrix for each frequency bin requested

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| X | C*16 | (*,*) | Input | First input complex matrix (N1,N2) |
| Y | C*16 | (*,*) | Input | Second input complex matrix (N2,N3) |
| N1 | I*4 | - | Input | First dimension of X and Z |
| N2 | I*4 | - | Input | Second dimension of X, first dimension of Y |
| N3 | I*4 | - | Input | Second dimension of Y and Z |
| Z | C*16 | (*,*) | Output | Output complex matrix (N1,N3) |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.7.4 Subroutine CMSCAL

CALLING SEQUENCE:

```
CMSCAL(X,Y,Z,N1,N2)
```

FUNCTION: This routine multiplies a complex matrix (Y) by a complex scalar (X) and returns the result in a complex matrix (Z)

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| X | C*16 | - | Input | Input complex scalar |
| Y | C*16 | (*,*) | Input | Input complex matrix (N1,N2) |
| N1 | I*4 | - | Input | First dimension of Y and Z |
| N2 | I*4 | - | Input | Second dimension of Y and Z |
| Z | C*16 | (*,*) | Output | Output complex matrix |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.7.2 Subroutine CMI2X2

CALLING SEQUENCE:

CMI2X2(A,AINV)

FUNCTION: Inverts a complex 2 by 2 matrix

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| A | C*16 | (2,2) | Input | Complex 2 by 2 matrix to be inverted |
| AINV | C*16 | (2,2) | Output | Resultant 2 by 2 complex inverted matrix |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.7.3 Subroutine CMMULT

CALLING SEQUENCE:

CMMULT(X,Y,Z,N1,N2,N3)

FUNCTION: This program multiplies two complex matrices and returns the result in a third matrix (Z)

| | |
|---|---|
| REALVEC | Computes components of complex matrix periodogram spectrum for i-th batch of data |
| RSLOPE | Removes zero-mean least-squares linear trend from data |
| SMEAN | Subtracts mean from data array |
| SMFFTS | Reads, smoothes, and plots power spectra and coherence |
| SMPLOTS | Computes the smoothed estimates plus and minus standard errors |
| STDEV | Computes standard deviation of batch of periodograms at each frequency |
| STNDE | Computes standard errors from standard deviations |
| WEIGHT | Computes data-weighting window |
| XIO | Calculates IO(X) |

## 3.7 GETDATA AND AR DETAILED SUBROUTINE DESCRIPTIONS

### 3.7.1 Subroutine CMCTR

CALLING SEQUENCE:

```
CMCTR(X,Y,N1,N2)
```

FUNCTION: Takes the complex conjugate transpose of X and places the result in Y

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| X | C*16 | (*,*) | Input | Input complex array |
| N1 | I*4 | - | Input | First dimension of X, second dimension of Y |
| N2 | I*4 | - | Input | Second dimension of X, first dimension of Y |
| Y | C*16 | (*,*) | Output | The complex conjugate transpose of X |

| | |
|---|---|
| JPLOT | Plots the power spectrum for each channel of data and the coherence between channels |
| JPLOT1 | Generates plots of the selected data |
| JREAD | Reads input data to the AR modeling program |
| JREMOV | Removes the slope from both channels found in X |
| JSUMRD | Calculates the sum-squared residuals |
| MMAXEN | Multivariate maximum entropy stepwise estimation of autoregression prediction matrices and one step prediction covariance matrix with order selection by Akaike's information criterion |
| PRTINF | Prints out an informational line |

## 3.6 FFT AND PLOTFFT SUBROUTINE DESCRIPTIONS

| | |
|---|---|
| AVED | Computes arithmetic mean of spectral quantities SD(F1).....SD(F2) |
| AVEE | Computes RMS of standard errors SE(F1)...SE(F2) |
| FFTKW | Computes FFT of time series λ using mixed-radix algorithm |
| FFTMR | Compute FFT by mixed-radix algorithm |
| FMEAN | Computes mean 2 by 2 complex periodogram |
| KWINDO | Multiplies two vectors of data by Kaiser window |
| LOGAV | Smoothes a spectrum using a constant-percent-bandwidth running mean |
| LXPRMT | Prompts the user for different plot options at the end of each frame |
| MSKPS | Computes coherence and cross-spectrum magnitude and phase |
| PPLOT | Plots the power spectrum and coherence |
| PRTINF | Prints an informational line of text |
| RDATA | Reads and decimates raw data one batch at a time |

```
@CL 'program name'    !compiles and loads into
                       PERIODLIB
@PLOTFFT              !links with main program and
                       creates new load module
```

Basic program information in a standard format is given below.

Name: PLOTFFT

Function: Plots power spectra and coherence from data files created by the program FFT

Common Blocks: *NONE*

Subprograms called: INPCHR,SMFFTS

## 3.5 GETDATA AND AR SUBROUTINE DESCRIPTIONS

| | |
|---|---|
| CMCTR | Takes the complex conjugate transpose of X and places the result in Y |
| CMI2X2 | Inverts a complex 2 by 2 matrix |
| CMMULT | Multiplies two complex matrices and returns the result in a third matrix (Z) |
| CMSCAL | Multiplies a complex matrix (Y) by a complex scalar (X) and returns the result in a complex matrix (Z) |
| CMZERO | Zeroes out a complex matrix |
| CPOWER | Calculates the power spectral density matrix for each frequency bin requested |
| DET | Evaluates determinant of an N by N matrix B |
| INVERT | Inverts N by N matrix B in place; ND is first dimension of B in calling program |
| J8REM | Removes slopes from both channels of input data X |
| JEQUAT | Equates matrix XX to matrix X |

| | | | | |
|---|---|---|---|---|
| A | R*8 | (*,*,*) | Output | Estimated partial forward auto regression matrices |
| B | R*8 | (*,*,*) | Output | Estimated partial backward autore-gression matrices |
| PF | R*8 | (*,*,*) | Output | Residual sum of squares and cross product matrices for each order fit. Last dimension of PF must be M+1 in calling program since order zero is stored in first position. |
| PB | R*8 | (*,*,*) | Output | Same for backward prediction |
| U | R*8 | (*,*) | Output | Estimated one step backward prediction covariance matrix |
| AICVEC | R*8 | (*) | Output | Vector containing AICs for each model searched |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: DET,INVERT

### 3.7.17 Subroutine PRTINF

CALLING SEQUENCE:

PRTINF(LINE)

FUNCTION: Prints out a informational line

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| LINE | CHAR | - | Input | Line to be printed |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

## 3.8 FFT AND PLOTFFT DETAILED SUBROUTINE DESCRIPTIONS

### 3.8.1 Subroutine AVED

CALLING SEQUENCE:

```
CALL AVED(F1,F2,SD,SMD,COUNT)
```

FUNCTION: Computes arithmetic mean of spectral quantities SD(F1).....SD(F2)

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| F1 | I*4 | - | Input | Minimum frequency bin number |
| F2 | I*4 | - | Input | Maximum frequency bin number |
| SD | R*4 | (*) | Input | Array of spectral quantities |
| COUNT | I*4 | - | Input | Index value |
| SMD | R*4 | (*) | Output | Arithmetic mean |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.2 Subroutine AVEE

CALLING SEQUENCE:

```
CALL AVEE(F1,F2,SE,SME,COUNT)
```

FUNCTION: Computes RMS of standard errors SE(F1)...SE(F2)

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| F1 | I*4 | - | Input | Minimum frequency bin number |
| F2 | I*4 | - | Input | Maximum frequency bin number |
| SE | R*4 | (*) | Input | Array of standard errors |
| COUNT | I*4 | - | Input | Index value |
| SME | R*4 | (*) | Output | RMS of standard errors |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.3 Subroutine FFTKW

CALLING SEQUENCE:

```
CALL FFTKW(X,NFREQ,NSIZE,RS,IS,IM)
```

FUNCTION: Computes FFT of time series X using mixed-radix algorithm

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| X | R*8 | (*) | Input | Array of time series data |
| NFREQ | I*4 | - | Input | Number of frequency bins |
| NSIZE | I*4 | - | Input | Number of data in time series |
| RS | R*8 | (*) | Output | Array of real parts of FFT of X |
| IS | R*8 | (*) | Output | Array of imaginary parts of FFT of X |
| IM | R*8 | (*) | Output | Dummy variable array |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: FFTMR

### 3.8.4 Subroutine FFTMR

CALLING SEQUENCE:

```
CALL FFTMR(A,B,NTOT,N,NSPAN,ISN)
```

FUNCTION: Compute FFT by mixed-radix algorithm

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| A | R*8 | (1) | Input | Real data array |
| B | R*8 | (1) | Input | Imaginary data array |
| NTOT | I*4 | - | Input | Number of complex data |
| N | I*4 | - | Input | Dimension of current variable |
| NSPAN | I*4 | - | Input | Algorithm parameter |
| ISN | I*4 | - | Input | Sign of complex exponential argument |
| A | R*8 | (1) | Output | Real data array |
| B | R*8 | (1) | Output | Imaginary data array |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.5 Subroutine FMEAN

CALLING SEQUENCE:

```
CALL FMEAN(NFREQ,RS12,MRS12,IS12,MIS12,S11,MS11,
          S22,MS22,A12,P12,K12,NBATCH,NSIZE,STEMP)
```

FUNCTION: Computes mean 2 by 2 complex periodogram

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NFREQ | I*4 | - | Input | Number of frequency bins |
| RS12 | R*8 | (*,*) | Input | Matrix of real cross-spectra |
| IS12 | R*8 | (*,*) | Input | Matrix of imaginary cross-spectra |
| S11 | R*8 | (*,*) | Input | Matrix of channel 1 autospectra |
| S22 | R*8 | (*,*) | Input | Matrix of channel 2 autospectra |
| NBATCH | I*4 | - | Input | Data batch number |
| NSIZE | I*4 | - | Input | Number of data per batch |
| MRS12 | R*8 | (*) | Output | Array of mean real cross-spectrum |
| MIS12 | R*8 | (*) | Output | Array of mean imaginary cross-spectrum |
| MS11 | R*8 | (*) | Output | Array of mean channel 1 auto-spectrum |
| MS22 | R*8 | (*) | Output | Array of mean channel 2 auto-spectrum |
| A12 | R*8 | (*) | Output | Array of magnitude of mean complex cross-spectrum |
| P12 | R*8 | (*) | Output | Array of phase of mean complex cross-spectrum |
| K12 | R*8 | (*) | Output | Array of spectral coherence |
| STEMP | R*4 | (*) | Output | Dummy array variable |

COMMON BLOCKS: NNTEST,TRIGS

SUBPROGRAMS CALLED: *NONE*

### 3.8.6 Subroutine KWINDO

CALLING SEQUENCE:

```
CALL KWINDO(N,X1,X2,WT,ALPH)
```

FUNCTION: Multiplies two vectors of data by Kaiser window

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| N | I*4 | - | Input | Number of data in each vector of data |
| X1 | R*8 | (*) | Input | First vector of data (array) |
| X2 | R*8 | (*) | Input | Second vector of data (array) |
| ALPH | R*8 | - | Input | Kaiser window alpha parameter |
| WT | R*8 | (*) | Output | Kaiser window (array) |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: WEIGHT

### 3.8.7 Subroutine LOGAV

CALLING SEQUENCE:

```
CALL LOGAV(NFREQ,FRACB,SD,SE,NCFREQ,CENTF,SMD,SME,NBATCH)
```

FUNCTION: Smoothes a spectrum using a constant-percent-bandwidth running mean

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NFREQ | I*4 | - | Input | Number of frequency bins |

| | | | | |
|---|---|---|---|---|
| FRACB | R*4 | - | Input | Fractional bandwidth parameter |
| SD | R*4 | (*) | Input | Estimated spectrum (array) |
| SE | R*4 | (*) | Input | Standard errors (array) |
| NCFREQ | I*4 | - | Output | Number of center frequencies |
| CENTF | R*4 | (*) | Output | Center frequency bin numbers (array) |
| SMD | R*4 | (*) | Output | Smoothed spectrum (array) |
| SME | R*4 | (*) | Output | Smoothed standard errors (array) |
| NBATCH | I*4 | - | Output | Number of data batches |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: AVED

### 3.8.8 Subroutine LXPRMT

CALLING SEQUENCE:

```
CALL LXPRMT(ICHG)
```

FUNCTION: Prompts the user for different plot options at the end of each frame

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| ICHG | I*4 | - | Output | Contains the code for the desired option |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.9 Subroutine MSKPS

CALLING SEQUENCE:

```
CALL MSKPS(NFREQ,MRS12,MIS12,MS12,K12,PS12,MS11,MS22)
```

FUNCTION: Computes coherence and cross-spectrum magnitude and phase

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NFREQ | I*4 | - | Input | Number of frequency bins |
| MRS12 | R*8 | (*) | Input | Mean real cross-spectrum array |
| MIS12 | R*8 | (*) | Input | Mean imaginary cross-spectrum array |
| MS11 | R*8 | (*) | Input | Mean autospectrum array for channel 1 |
| MS22 | R*8 | (*) | Input | Mean autospectrum array for channel 2 |
| MS12 | R*8 | (*) | Output | Magnitude of mean cross-spectrum array |
| K12 | R*8 | (*) | Output | Coherence array |
| PS12 | R*8 | (*) | Output | Phase array |

COMMON BLOCKS: NNTEST

SUBPROGRAMS CALLED: *NONE*

### 3.8.10 Subroutine PPLOT

CALLING SEQUENCE:

```
CALL PPLOT(NFREQ,NCFREQ,FREQ,T1,T2,XMIN,XMAX,YMIN,YMAX,
LTYPE,NBATCH,IFIRST,NSTRT,HEADER,HDINFO,IHD,ENDFLG,NPLOTS)
```

FUNCTION: This routine plots the power spectrum and coherence

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NFREQ | I*4 | - | Input | Number of frequency bins |
| NCFREQ | I*4 | - | Input | Number of center frequencies |
| FREQ | R*4 | (*) | Input | Array of frequencies |
| T1 | R*4 | (4097,3) | Input | Power spectrum or coherence array to be plotted |
| T2 | R*4 | (4097,3) | Input | Channel 2 power spectrum to be plotted |
| XMIN | R*4 | - | Input | Default X-axis minimum |
| XMAX | R*4 | - | Input | Default X-axis maximum |
| YMIN | R*4 | - | Input | Default Y-axis minimum |
| YMAX | R*4 | - | Input | Default Y-axis maximum |
| LTYPE | I*4 | - | Input | Default X and Y axis type |
| NBATCH | I*4 | - | Input | Number of batches of data |
| IFIRST | I*4 | - | Input | Automatic scaling flag |
| NSTRT | I*4 | - | Input | Starting data point to be plotted |
| HEADER | CHAR | - | Input | Plot header line |
| HDINFO | CHAR | (5) | Input | Plot descriptive information to be plotted at the bottom of the frame |
| IHD | I*4 | - | Input | Number of lines of HDINFO |
| NPLOTS | I*4 | - | Input | Flag for power spectrum or coherence plots |
| XMIN | R*4 | - | Output | Current default X-axis minimum |
| XMAX | R*4 | - | Output | Current X-axis maximum |
| YMIN | R*4 | - | Output | Current Y-axis minimum |

| | | | | |
|---|---|---|---|---|
| YMAX | R*4 | - | Output | Current Y-axis maximum |
| LTYPE | I*4 | - | Output | Current X and Y axis types |
| IFIRST | I*4 | - | Output | Automatic scaling flag |
| ENDFLG | R*4 | - | Output | Program termination flag |

COMMON BLOCKS: WSPACE

SUBPROGRAMS CALLED: AGSETP,INPRL4,DISPLA,ANOTAT,AGSETF,AGSETI, EZMXY,PWRY,FRAME

### 3.8.11 Subroutine PRTINF

CALLING SEQUENCE:

```
CALL PRTINF(LINE)
```

FUNCTION: Prints an informational line of text

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| LINE | CHAR | - | Input | Line to be printed |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.12 Subroutine RDATA

CALLING SEQUENCE:

```
CALL RDATA(X1,X2,NSIZE,NDEC,IERR,ENDFLG)
```

FUNCTION: Reads and decimates raw data one batch at a time

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NSIZE | I*4 | - | Input | Size of batch to be read |
| NDEC | I*4 | - | Input | Decimation factor |
| X1 | R*8 | (*) | Output | Channel 1 decimated data |
| X2 | R*8 | (*) | Output | Channel 2 decimated data |
| IERR | I*4 | - | Output | Error during read flag |
| ENDFLG | L*1 | - | Output | End of file flag |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.13 Subroutine REALVEC

CALLING SEQUENCE:

```
CALL REALVEC(NBATCH,RS1,RS2,IS1,IS2,S11,S22,RS12,IS12,
             NFREQ,I,NSIZE)
```

FUNCTION: Computes components of complex matrix periodogram spectrum for i-th batch of data

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NBATCH | I*4 | - | Input | Number of batches |
| RS1 | R*8 | (*) | Input | Real part of channel 1 FFT array |
| RS2 | R*8 | (*) | Input | Real part of channel 2 FFT array |
| IS1 | R*8 | (*) | Input | Imaginary part of channel 1 FFT array |
| IS2 | R*8 | (*) | Input | Imaginary part of channel 2 FFT array |
| NFREQ | I*4 | - | Input | Number of frequency bins |
| I | I*4 | - | Input | Batch number |

| | | | | |
|---|---|---|---|---|
| NSIZE | I*4 | - | Input | Number of data per batch |
| S11 | R*8 | (*,*) | Output | Channel 1 autospectrum matrix |
| S22 | R*8 | (*,*) | Output | Channel 2 autospectrum matrix |
| RS12 | R*8 | (*,*) | Output | Real part of cross-spectrum matrix |
| IS12 | R*8 | (*,*) | Output | Imaginary part of cross-spectrum matrix |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.14 Subroutine RSLOPE

CALLING SEQUENCE:

```
CALL RSLOPE(NSIZE,X,SLOPE)
```

FUNCTION: Removes zero-mean least-squares linear trend from data

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NSIZE | I*4 | - | Input | Number of data |
| X | R*8 | (*) | Input | Data array |
| SLOPE | R*4 | - | Output | Slope of linear trend |
| X | R*8 | (*) | Output | Data array |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.15 Subroutine SMEAN

CALLING SEQUENCE:

```
CALL SMEAN(NSIZE,X)
```

FUNCTION: Subtracts mean from data array

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NSIZE | I*4 | - | Input | Number of data |
| X | R*8 | (*) | Input | Data array |
| X | R*8 | (*) | Output | Data array with mean subtracted |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.16 Subroutine SMFFTS

CALLING SEQUENCE:

```
CALL SMFFTS(NFREQ,NBATCH,FSAMP,CENTF,FREQ,FRACB,
           S1,SDS1,SES1,SMS1,SMES1,

           S2,SDS2,SES2,SMS2,SMES2,

           A12,SDA12,SEA12,SMA12,SMEA12,

           P12,SDP12,SEP12,SMP12,SMEP12,

           K12,SDK12,SEK12,SMK12,SMEK12,

           TS1,TS2,TA12,TP12,TK12,HEADER,HDINFO,IHD)
```

FUNCTION: Reads, smoothes, and plots power spectra and coherence

RGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NFREQ | I*4 | - | Input | Number of frequency bins |
| NBATCH | I*4 | - | Input | Number of batches of data |
| FSAMP | R*4 | - | Input | Sampling frequency of time series data |
| CENTF | R*4 | (*) | Input | Array of frequency bin numbers of center frequencies |
| FREQ | R*4 | (*) | Input | Array of frequencies |
| FRACB | R*4 | - | Input | Fractional bandwidth parameter |
| S1 | R*4 | (*) | Input | Autospectrum array for channel 1 |
| SDS1 | R*4 | (*) | Input | Standard deviation of auto spectrum for channel 1 (array) |
| S2 | R*4 | (*) | Input | Autospectrum array for channel 2 |
| SDS2 | R*4 | (*) | Input | Standard deviation of auto spectrum for channel 2 (array) |
| A12 | R*4 | (*) | Input | Magnitude of cross-spectrum (array) |
| SDA12 | R*4 | (*) | Input | Standard deviation of cross-spectrum (array) |
| P12 | R*4 | (*) | Input | Magnitude of phase (array) |
| SDP12 | R*4 | (*) | Input | Standard deviation of phase (array) |
| K12 | R*4 | (*) | Input | Magnitude of coherence (array) |
| SDK12 | R*4 | (*) | Input | Standard deviation of coherence (array) |
| TS1 | R*4 | (4097,3) | Input | Channel 1 overlay of SMS1, plus and minus standard error |
| TS2 | R*4 | (4097,3) | Input | Channel 2 overlay of SMS2, plus and minus standard error |

| | | | | |
|---|---|---|---|---|
| TA12 | R*4 | (4097,3) | Input | Magnitude overlay of SMA12, plus and minus standard error |
| TP12 | R*4 | (4097,3) | Input | Phase overlay of SMP12, plus and minus standard error |
| TK12 | R*4 | (4097,3) | Input | Coherence overlay of SMK12, plus and minus standard error |
| HEADER | CHAR | - | Input | Plot header line |
| HDINFO | CHAR | (5) | Input | Plot data identification lines written at the bottom of all plots |
| IHD | I*4 | - | Input | Number of HDINFO lines |
| SES1 | R*4 | (*) | Output | Standard errors of autospectrum for channel 1 (array) |
| SMS1 | R*4 | (*) | Output | Smoothed autospectrum array for channel 1 (array) |
| SMES1 | R*4 | (*) | Output | Standard errors of smoothed autospectrum for channel 1 (array) |
| SES2 | R*4 | (*) | Output | Standard errors of autospectrum for channel 2 (array) |
| SMS2 | R*4 | (*) | Output | Smoothed autospectrum array for channel 2 (array) |
| SMES2 | R*4 | (*) | Output | Standard errors of smoothed autospectrum for channel 2 (array) |
| SEA12 | R*4 | (*) | Output | Standard errors of cross-spectrum (array) |
| SMA12 | R*4 | (*) | Output | Smoothed magnitude of cross-spectrum (array) |
| SMEA12 | R*4 | (*) | Output | Standard errors of smoothed magnitude of cross-spectrum (array) |

| | | | | |
|---|---|---|---|---|
| SEP12 | R*4 | (*) | Output | Standard errors of cross-spectrum (array) |
| SMP12 | R*4 | (*) | Output | Smoothed cross-spectrum phase (array) |
| SMEP12 | R*4 | (*) | Output | Standard errors of smoothed cross-spectrum phase (array) |
| SEK12 | R*4 | (*) | Output | Standard errors of coherence (array) |
| SMK12 | R*4 | (*) | Output | Smoothed coherence (array) |
| SMEK12 | R*4 | (*) | Output | Standard errors of smoothed coherence (array) |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: STNDE,LOGAV,SMPLOTS,INPRL4

### 3.8.17 Subroutine SMPLOTS

CALLING SEQUENCE:

```
CALL SMPLOTS(NFREQ,NCFREQ,NBATCH,CENTF,FSAMP,FREQ,
        SMES1,SMES2,SMEA12,SMEP12,SMEK12,
        TS1,TS2,TA12,TP12,TK12,HEADER,HDINFO,IHD,ENDFLG)
```

FUNCTION: Computes the smoothed estimates plus and minus standard errors

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NFREQ | I*4 | - | Input | Number of frequency bins |
| NCFREQ | I*4 | - | Input | Number of center frequencies |
| NBATCH | I*4 | - | Input | Number of batches of data |

| | | | | |
|---|---|---|---|---|
| CENTF | R*4 | (*) | Input | Array of frequency bin numbers of center frequencies |
| FSAMP | R*4 | - | Input | Sampling frequency of time series data |
| FREQ | R*4 | (*) | Input | Array of frequencies |
| SMES1 | R*4 | (*) | Input | Standard errors of smoothed autospectrum for channel 1 (array) |
| SMES2 | R*4 | (*) | Input | Standard errors of smoothed autospectrum for channel 2 (array) |
| SMEA12 | R*4 | (*) | Input | Standard errors of smoothed magnitude of cross-spectrum (array) |
| SMEP12 | R*4 | (*) | Input | Standard errors of smoothed cross-spectrum phase (array) |
| SMEK12 | R*4 | (*) | Input | Standard errors of smoothed coherence (array) |
| TS1 | R*4 | (4097,3) | Input | Channel 1 overlay of SMS1, plus and minus standard error |
| TS2 | R*4 | (4097,3) | Input | Channel 2 overlay of SMS2, plus and minus standard error |
| TA12 | R*4 | (4097,3) | INPUT | Magnitude overlay of SMA12, plus and minus standard error |
| TP12 | R*4 | (4097,3) | INPUT | Phase overlay of SMP12, plus and minus standard error |
| TK12 | R*4 | (4097,3) | INPUT | Coherence overlay of SMK12, plus and minus standard error |
| HEADER | CHAR | - | INPUT | Plot header line |
| HDINFO | CHAR | (5) | INPUT | Plot data identification lines written at the bottom of all plots |

| | | | | |
|---|---|---|---|---|
| IHD | I*4 | - | INPUT | Number of HDINFO lines |
| ENDFLG | I*4 | - | OUTPUT | Program termination flag |

COMMON BLOCKS: WSPACE

SUBPROGRAMS CALLED: AGGETP,PPLOT,INPIN4

### 3.8.18 Subroutine STDEV

CALLING SEQUENCE:

```
CALL STDEV(NFREQ,RS12,MRS12,IS12,MIS12,S11,MS11,S22,MS22,
          A12,SDS11,SDS22,SDA12,SDP12,SDK12,K12,NBATCH,SDIS12,
          SDRS12,STEMP)
```

FUNCTION: Computes standard deviation of batch of periodograms at each frequency

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NFREQ | I*4 | - | Input | Number of frequency bins |
| RS12 | R*8 | (*,*) | Input | Real parts of cross-spectra (matrix) |
| MRS12 | R*8 | (*) | Input | Mean of real cross-spectra (array) |
| IS12 | R*8 | (*,*) | Input | Imaginary parts of cross-spectra (array) |
| MIS12 | R*8 | (*) | Input | Mean imaginary cross-spectra (matrix) |
| S11 | R*8 | (*,*) | Input | Autospectra for channel 1 (matrix) |
| MS11 | R*8 | (*) | Input | Mean autospectrum for channel 1 (array) |

| | | | | |
|---|---|---|---|---|
| S22 | R*8 | (*,*) | Input | Autospectra for channel 2 (matrix) |
| MS22 | R*8 | (*) | Input | Mean autospectra for channel 2 (array) |
| A12 | R*8 | (*) | Input | Magnitude of cross-spectrum (array) |
| K12 | R*8 | (*) | Input | Coherence (array) |
| NBATCH | I*4 | - | Input | Number of data batches |
| STEMP | R*4 | (*) | Input | Real * 4 disk storage temporary array |
| SDS11 | R*8 | (*) | Output | Standard deviations of channel 1 autospectra (array) |
| SDS22 | R*8 | (*) | Output | Standard deviations of channel 2 autospectra (array) |
| SDA12 | R*8 | (*) | Output | Standard deviations of cross-spectrum magnitude (array) |
| SDP12 | R*8 | (*) | Output | Standard deviation of cross-spectrum phase (array) |
| SDK12 | R*8 | (*) | Output | Standard deviations of coherence (array) |
| SDIS12 | R*8 | (*) | Output | Standard deviations of imaginary cross-spectra (array) |
| SDRS12 | R*8 | (*) | Output | Standard deviations of real cross-spectra (array) |

COMMON BLOCKS: TRIGS,NNTEST

SUBPROGRAMS CALLED: *NONE*

### 3.8.19 Subroutine STNDE

CALLING SEQUENCE:

```
CALL STNDE(NFREQ,SD,SE,NBATCH)
```

FUNCTION: Computes standard errors from standard deviations

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| NFREQ | I*4 | - | Input | Number of frequency bins |
| SD | R*4 | (*) | Input | Array of standard deviations |
| NBATCH | I*4 | - | Input | Number of batches of data |
| SE | R*4 | (*) | Output | Array of standard errors |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.20 Subroutine WEIGHT

CALLING SEQUENCE:

```
CALL WEIGHT(WT,N,UF,SF,IOPT,ALPH)
```

FUNCTION: Computes data-weighting window

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| N | I*4 | - | Input | Number of coefficients |
| IOPT | I*4 | - | Input | Window selection parameter |
| ALPH | R*8 | - | Input | Window parameter |
| WT | R*8 | (1) | Output | Array of weighting coefficients |
| UF | R*8 | - | Output | Sum of squares of coefficients |
| SF | R*8 | - | Output | Scale factor |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

### 3.8.21 Function XIO

CALLING SEQUENCE:

```
RX = XIO(X)
```

FUNCTION: Calculates $IO(X) = 1+\sum_{K=1}^{\infty} (((X/2)^{**}K/FAC(K))^{**}2)$
where FAC is factorial

ARGUMENTS:

| NAME | TYPE | DIMENSION | IN/OUT | DESCRIPTION |
|---|---|---|---|---|
| X | R*8 | - | Input | Function value to be evaluated |
| XIO | R*8 | - | Output | Output of function |

COMMON BLOCKS: *NONE*

SUBPROGRAMS CALLED: *NONE*

## REFERENCES

1. White, J.V., "Error Models for Gravity Gradiometers in Airborne Surveys," Technical Report AFGL-TR-80-0220, Air Force Geophysics Laboratory, Hanscom AFB, January 1980. ADA097745.

END

FILMED

7-85

DTIC